

RECEIVED
CENTRAL FAX CENTER

AUG 18 2006

APPLICANT(S): GAT, Tal et al.
SERIAL NO.: 10/748,173
FILED: December 31, 2003
Page 2

AMENDMENTS TO THE CLAIMS

Please add or amend the claims to read as follows, and cancel without prejudice or disclaimer claims indicated as cancelled. The following listing of claims is intended to replace all prior versions and/or listings of claims in the Application:

LISTING OF CLAIMS

1. (Currently Amended) A method comprising:
generating branch predictions for two sequential lines in parallel during a prediction period;
storing [[a]] said branch predictions in a queue; and
delivering [[said]] a stored branch prediction from said queue to an instruction fetch unit.
2. (Currently Amended) The method as in claim 1, ~~comprising generating said branch prediction for two sequential lines in~~ wherein said prediction period comprises two clock cycles.
3. (Original) The method as in claim 1, comprising segmenting a cache of a branch predictor into a first side and a second side, where entries on said first side correspond to addresses having even-numbered indexes, and entries on said second side correspond to addresses having odd-numbered indexes.
4. (Original) The method as in claim 3, wherein an index of one of two sequential lines corresponds to an entry on said first side of said cache, and an index of another of said two sequential lines corresponds to an entry on said second side of said cache.
5. (Cancelled) ~~The method as in claim 1, comprising, in two cycles, storing said prediction for two sequential lines in said queue and delivering said prediction to said instruction fetch unit.~~
6. (Original) The method as in claim 1, comprising generating branch predictions for a stream of addresses during a stall of said instruction fetch unit.

APPLICANT(S): GAT, Tal et al.
SERIAL NO.: 10/748,173
FILED: December 31, 2003
Page 3

7. (Original) The method as in claim 1, comprising generating during a cycle a prediction for a line, said line being other than the line being fetched by said instruction fetch unit during said cycle.

8. (Original) The method as in claim 1, wherein the addresses for which predictions are generated by a branch prediction unit are decoupled from the addresses for which lines are fetched by said instruction fetch unit.

9. (Cancelled) A processor comprising:

~~a branch prediction unit; and~~

~~an instruction fetch unit;~~

~~wherein said branch prediction unit is to, in a prediction period, generate a prediction on a first line, and said instruction fetch unit is to in said prediction period, fetch instructions for a second line.~~

10. (Cancelled) The processor as in claim 9, ~~wherein said branch prediction unit comprises a queue configured to store branch predictions.~~

11. (Currently Amended) The ~~[[processor]]~~ method as in claim ~~[[9]]~~ 1, ~~wherein said branch prediction unit is to deliver~~ comprising delivering a branch prediction to said instruction fetch unit in the same prediction period as said branch prediction unit ~~writes said branch prediction to a~~ is written to said queue.

12. (Currently Amended) A processor comprising:

an instruction fetch unit; and

a branch prediction unit, said branch prediction unit comprising a queue to store branch predictions, said branch prediction unit to generate branch predictions for two sequential lines in parallel during a prediction period, and said branch prediction unit to deliver branch predictions stored in said queue to said instruction fetch unit.

13. (Original) The processor as in claim 12, wherein said branch prediction unit comprises a cache whose entries are segmented into a first side and a second side, where entries on said

APPLICANT(S): GAT, Tal et al.
SERIAL NO.: 10/748,173
FILED: December 31, 2003
Page 4

first side correspond to addresses having even-numbered indexes, and entries on said second side correspond to addresses having odd-numbered indexes.

14. **(Currently Amended)** The processor as in claim ~~[[12]]~~ 13, wherein said cache is to store odd-numbered addresses in a first segment of said cache, and even-numbered addresses in a second segment of said cache.

15. **(Currently Amended)** The processor as in claim 12, wherein said ~~branch prediction unit is to look up two lines in a~~ prediction period comprises two clock cycles.

16. **(Original)** The processor as in claim 12, wherein said branch prediction unit is, in a prediction period, to write a branch prediction to said queue and to deliver said branch prediction to said instruction fetch unit.

17. **(Cancelled)** A method comprising:
generating during a cycle, in a branch prediction unit of a processor, a branch prediction for a first line; and
fetching during said cycle, in an instruction fetch unit of said processor, an instruction for a second line.

18. **(Cancelled)** A method as in claim 17, comprising storing said branch prediction in a data storage area of said branch prediction unit.

19. **(Cancelled)** A method as in claim 17, comprising segmenting a cache of said branch prediction unit into a first side and a second side, where entries on said first side correspond to addresses having even-numbered indexes, and entries on said second side correspond to addresses having odd-numbered indexes.

20. **(Currently Amended)** A system comprising:
a dynamic random access memory unit; and
a processor comprising:
an instruction fetch unit; and

APPLICANT(S): GAT, Tal et al.
SERIAL NO.: 10/748,173
FILED: December 31, 2003
Page 5

a branch prediction unit, said branch prediction unit comprising a queue to store branch predictions, said branch prediction unit to generate branch predictions for two sequential lines in parallel during a prediction period, and said branch prediction unit to deliver branch predictions stored in said queue to said instruction fetch unit.

21. (Original) A system as in claim 20, wherein said branch prediction unit comprises a cache divided into a first side and a second side, where entries on said first side correspond to addresses having even-numbered indexes, and entries on said second side correspond to addresses having odd-numbered indexes.
22. (Original) A system as in claim 21, wherein said cache is configured to store odd addresses in said odd side, and even addresses in said even side.
23. (New) The system as in claim 20, wherein said prediction period comprises two clock cycles.
24. (New) The system as in claim 20, wherein said branch prediction unit is, in a prediction period, to write a branch prediction to said queue and to deliver said branch prediction to said instruction fetch unit.
25. (New) The system as in claim 20, wherein the addresses for which predictions are generated by said branch prediction unit are decoupled from the addresses for which lines are fetched by said instruction fetch unit.